Acquisition of Software Intensive Systems Conference

Revitalizing the Software Acquisition Process

28 January 2003

Mike Nicol
ASC/EN

Air Force Aeronautical Systems Center, Engineering Directorate

Outline

ASC

- **Our History**
- **Today's Environment**
- **Plan for Improvement**
- **Next Steps**
- **Conclusions**

A Decade Ago...

ASC

- **DoD 5000.2, Part 6-D, *Computer Resources***
  - Computer Resources Life-Cycle Management Plan (CRLCMP), Integrated system development, Software metrics, Software test management, Ada language policy, Software engineering practices
- **Air Force Regulation (AFR) 800-14, Life Cycle Management of Computer Resources in Systems**
- **AFMC Pamphlets**
  - Software IV&V, Software Risk Abatement, Review of Software Requirements and Interface Requirements Specifications, Software Management Indicators, Software Quality Measurement, Software Development Capability Assessment
- **SAF/AQ Memos**
  - Software Engineering, Software Maturity Assessment, Ada, Metrics, Software Estimating, Software Reuse, Best Practices, Use of Software Development Capability Evaluation in Source Selection, Etc.

A Decade Ago...(Cont.)

ASC

- **Development standards**
  - DOD-STD-2167/2168, MIL-STD-498, MIL-STD-1803
  - MIL-STD-882, MIL-STD-490, MIL-STD-499, DOD-STD-1521
- **Senior software engineer in each program office, supported with additional help, as necessary**
  - Depending on magnitude of software development effort, program phase, etc.
- **Air Force Systems Acquisition School training**
  - Computer Resources Acquisition Course (CRAC)

In spite of all this, success was not guaranteed...

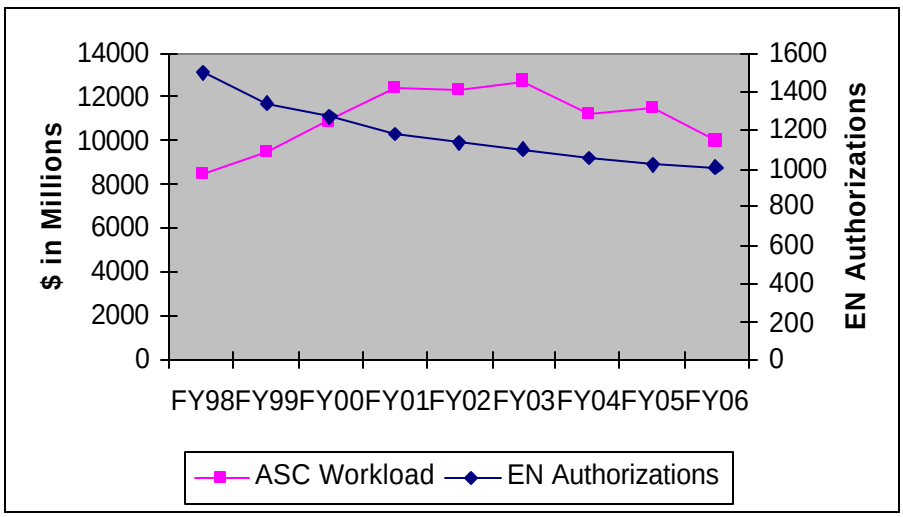
Today...

ASC

- **Limited policy / guidance specific to the acquisition of software intensive systems**
  - Almost none of it mandatory
- **No standard way of doing business**
  - Processes across the acquisition enterprise have diverged
  - Decreasing oversight / insight
- **Lack of appreciation for process**
  - Demands for reduced cycle time
- **Training available through SAM courses**
  - Data indicates limited exposure
- **Aging and diminishing workforce**
  - 10 year gap for new hires
  - Acquisition workforce being rapidly downsized

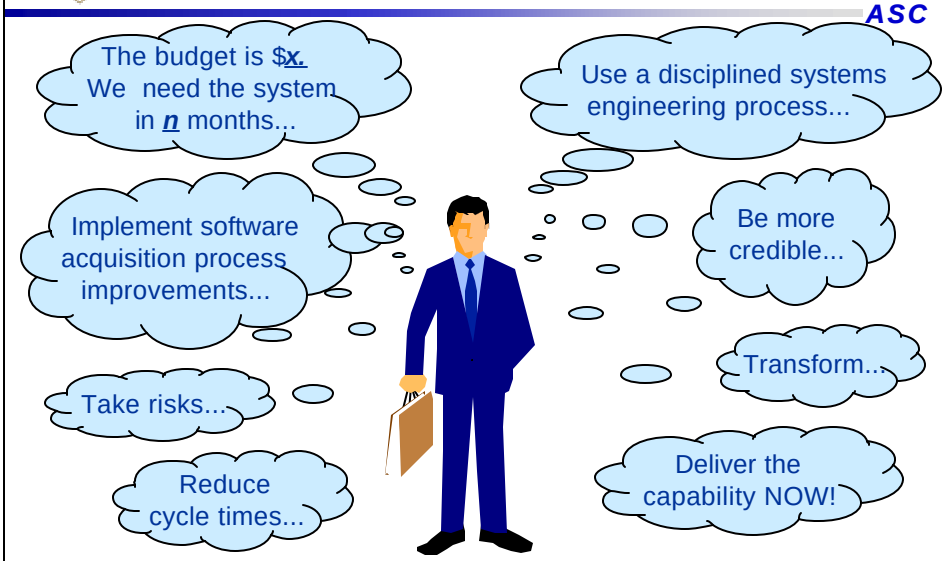
ASC Workload and EN Staffing

ASC



The Message(s) to PMs...

ASC



A Sample of Findings from ASC Program Reviews

ASC

- Incompatible / optimistic performance, effort, and schedule baselines
- Deficiencies in requirements management
- Inadequate risk identification and management
- Processes set aside due to program pressures
- Planned reuse not achieved
- Program staffing problems
- Failure to identify and react to problems
- Inadequate program office insight
- Labs not fully capable or not in place when needed
- Fixed price development contracts with uncertain requirements or other significant risks

An Issue Seen Too Frequently…

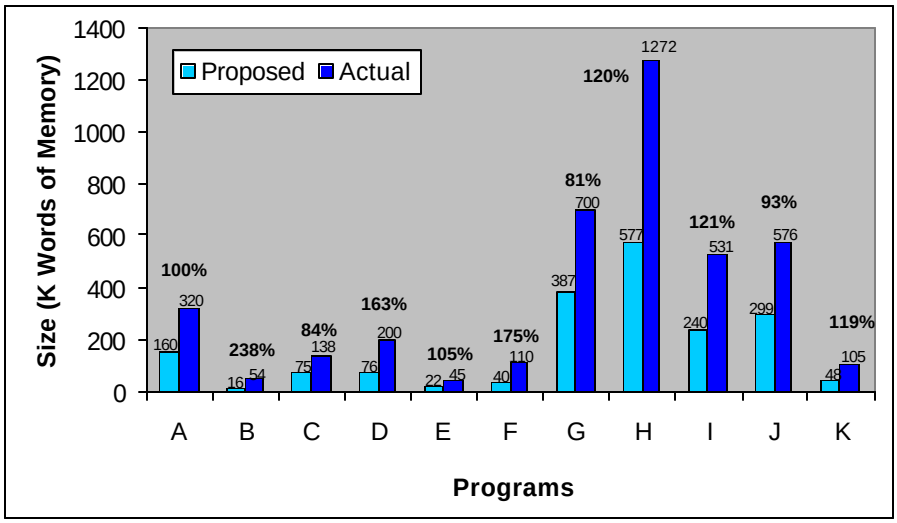
ASC

Inability to establish compatible effort, schedule, and performance baselines

- **Why?**
  - Programs come with defined cost, schedule, and performance baselines, often (optimistically) determined without adequate insight into what actually needs to be done
  - All participants challenged to reduce cycle times, take risks, etc.
  - Requirements are not fully defined / stable
  - Difficult to estimate the size of a software development effort for unprecedented systems or where requirements are not complete
    - Hence, difficult to estimate development effort and schedule
  - History indicates software size estimate grows significantly during development

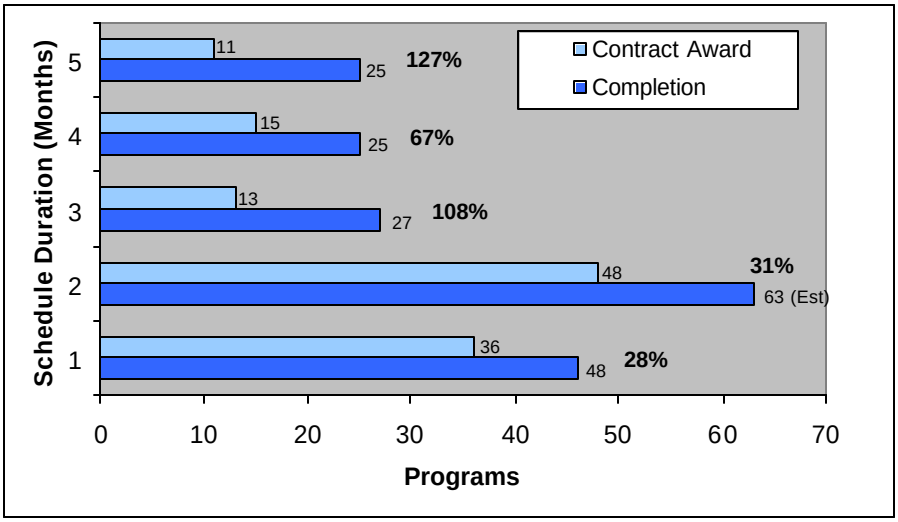
Embedded Software Size Growth

ASC



Embedded Software Schedule Growth

ASC



Plan for Improvement

ASC

- **ASC/EN initiative to document and improve systems engineering processes**
  - Identify, define, & document technical processes
    - Combine and simplify current processes
    - Fully integrate internal processes
    - Focus on government responsibilities
  - Deploy
    - Training, guidance, and monitoring
- **Also considering independent look**
  - Validation of selected processes by outside organization

Software Acquisition Approach

ASC

- **Document critical processes**
  - Enterprise support activities
  - Program-level activities (planning and execution)
- **Develop training**
  - Target to all who need to know - not just organic engineering
- **Deploy and monitor application of processes**

Software Acquisition Approach (Cont.)

ASC

- **Organization**
  - Enterprise (Center Level) Support (3)
  - Acquisition Program Planning Processes (4)
  - Acquisition Program Execution Processes (14)
- **Key practices addressed in acquisition strategy**
- **Each process documented with brief description**
  - Purpose
  - Roles and Responsibilities
  - Key steps
  - Inputs
  - Outputs / Products
  - Available Tools / Techniques
  - Potential Problem Areas / Pitfalls
  - Lessons Learned
- **Appendices with additional detail as needed**

Key Software Acquisition Practices
(To Be Addressed in Acquisition Strategy)

ASC

- Establish Realistic and Compatible Program Baselines
- Provide System Development and Demonstration (SDD) Phase Source Selection Support
- Identify and Manage Computer System and Software Risks
- Establish and Manage Software Requirements
- Accommodate High-Assurance Systems
- Ensure Application of Mature Development Processes
- Maintain Technical Insight

Enterprise Support

ASC

- **Provide Advice and Counsel**
  - Pre-Acquisition Strategy Panel (ASP) Support
  - Source Selection Consultation and Advice
  - Ensure achievable baselines
  - Program Execution
  - Independent Reviews
- **Manage Software Acquisition Training and Experience**
  - Software Acquisition Engineering Training (Guidebook)
  - Software Estimation Training
  - Other special-topic training as required
- **Collect and Disseminate Lessons Learned**
  - Collect lessons learned at project/build completion
  - Establish and maintain lessons learned database
  - Disseminate lessons learned through briefings, training, etc.
  - Implement needed process improvements

Acquisition Program Planning

ASC

3.1 Develop Software Acquisition Strategy

- Develop program approach to key software acquisition practices
- Get informal, independent review prior to issuing RFP

3.2 Establish Realistic, Compatible Program Baselines

- Estimate software development size (factoring in growth)
- Estimate software development effort and schedule
- Develop realistic estimate that balances risk, cycle time, etc.

3.3 Support Request for Proposal (RFP) Preparation

- Provide key software considerations for RFP Sections L and M
- Solicit and evaluate software size, effort, and schedule estimates
- Solicit software development process documentation

3.4 Provide Source Selection Support

- Evaluate developer capability
- Evaluate proposed development processes
- Evaluate proposed development plan
- Assess compatibility of processes, plan, effort, and schedule

Program Level Execution

ASC

4.1 Identify and Manage Software-Related Risks

- Ensure effective risk management process is in place
- Ensure all software-related risks are identified and managed

4.2 Establish and Manage Software Requirements

- Ensure software requirements are defined, complete, verified, consistent and traceable

4.3 Address Training System Concurrency Requirements

- Provide for the most efficient method to meet training system concurrency requirements

4.4 Establish Software Build Plan

- Ensure there is a plan to define, develop, integrate, and deliver software increments in response to system requirements

Program Level Execution (Cont.)

ASC

4.5 Accommodate Application and Sustainment of Non-Developmental Software (NDS)

- Address COTS and other NDS integration

4.6 Accommodate Security Certification & Accreditation (C&A)

- Ensure that confidentiality, integrity, and availability is maintained throughout the life-cycle of the system
- Preclude compromise, exploitation, sabotage, and intentional damage and destruction

4.7 Accommodate Safety-Critical and High-Assurance Systems

- Define the process, including what is expected of the developer, to specify, design, develop, integrate, and verify flight-critical and safety-critical systems

Program Level Execution (Cont.)

ASC

4.8 Establish System / Software Engineering Environment (S/SEE) and Development and Integration Laboratories

- Ensure development, integration, and verification environment requirements are fully defined
- Ensure environments are in place when needed and can provide the required throughput

4.9 Ensure Application of Mature Development Processes

- Assess developer team process capability prior to contract award to identify strengths, weaknesses, and risks
- Support disciplined application of processes throughout the development effort

Program Level Execution (Cont.)

ASC

4.10 Maintain Technical Insight and Resolve Development Issues

- Implement effective means of communication on program status and issues
- Take corrective action when necessary

4.11 Establish Software Product Engineering Data

- Ensure the minimum set of engineering data and documentation required for the weapon system software is developed, acquired (or escrowed), and maintained

4.12 Conduct / Support Technical Reviews

- Determine the types of reviews to be accomplished and the role of the acquisition organization
- Establish relevant entry and exit criteria

Program Level Execution (Cont.)

ASC

4.13 Plan for Post Deployment Software Support

- Identify source of support for all software elements
- Determine expected rates of change and expected workload
- Establish required support resources and facilities

4.14 Identify and Collect Lessons Learned

- Survey project participants
- Collect objective data
- Share the results

Process Development Schedule

ASC

Complete guidebook draft	January 2003
Complete coordination and review by SPOs and AFMC SISSG members	February 2003
Publish guidebook Version 1	February 2003
Complete development of guidebook training	March 2003

Next Steps

ASC

- Consider extending scope to AFMC
  - Address concerns of other domains
  - Add sustainment processes
- Get leadership buy-in
- Work with Air Force Institute of Technology (AFIT) to enhance training
  - Software Professional Development Program (SPDP)
  - Acquisition-specific training
- Address Section 804 requirements
- Incorporate improvements identified by independent process validation activities

Conclusions

ASC

- **We understand the issues and are taking positive steps to set programs up to succeed**
- **Revitalizing our processes is a crucial first step**
- **We can't solve the problem by ourselves**
  - Balance risk and credibility
  - Support disciplined application of processes